WHAT IS CLAIMED IS:

Sub A1

1. A distributed multiprocessor server system for facilitating delegated processing of at least portions of requests associated with request messages received via a communicatively coupled network link, the system comprising:

a network interface;

an intelligent switch coupled to the network interface, the switch comprising logic components for identifying a new request, corresponding to a message packet received by the network interface, passed from the network interface to the intelligent switch;

a default handler processor coupled to the intelligent switch and configured to receive the new request from the intelligent switch, the default handler processor comprising delegation logic facilitating: associating a request type with at least a portion of the new request, identifying a handler processor from a set of specialized handler processors for executing at least the portion of the new request based upon the request type, and issuing a message reassigning at least the portion of the new request to the identified handler processor; and

at least one bus structure communicatively linking the set of specialized handler processors to the intelligent switch and request reassignment tracking logic enabling the intelligent switch to route messages associated with at least the portion of the reassigned request between the identified handler processor and the network interface, thereby facilitating completing at least the portion of the new request through communications between the identified handler of specialized handler processors and the network interface via the intelligent switch without intervention by the default handler processor.

2. The distributed multiprocessor server system of claim 1 further comprising a storage server system linked to the intelligent switch via a non-blocking switch.

3. The distributed multiprocessor server system of claim 2 wherein the storage server system comprises memory arranged as a set of version controlled partitions.

4. The distributed multiprocessor server system of claim 3 wherein a straddle is incorporated into a partition, thereby facilitating continuous availability of all stored data while a particular partition is relocated within the storage server system.

5. The distributed multiprocessor server system of claim 2 wherein a state of a file maintained by the storage server system is represented in the form of a bitmap entry; and wherein a first bit is associated with a creator of new data in the file and a second bit is associated with a deleter of data stored in the file.

6. The distributed multiprocessor server system of claim 2 wherein the intelligent switch receives messages from the non-blocking switch in the form of ATM cells.

7. The distributed multiprocessor server system of claim 2 wherein the set of specialized handler processors includes a processor facilitating transfer of a file stored on the storage server system.

8. The distributed multiprocessor server system of claim 7 wherein the file transfer is performed in accordance with a TCP named file transfer protocol over an identified connection.

9. The distributed multiprocessor server system of claim 2 wherein the set of specialized handler processors includes a processor including functionality facilitating transforming the data within a file prior to transfer.

A2

10. The distributed multiprocessor server system of claim 1 wherein the set of specialized handler processors includes a processor including computer gateway interface (CGI) functionality.

11. The distributed multiprocessor server system of claim 1 further comprising a data retrieval buffer interposed between a data storage facility and the set of specialized handler processors, the data retrieval buffer being independently accessible with respect to a primary RAM utilized by the default handler processor.

12. The distributed multiprocessor server system of claim 1 further comprising new/old request differentiation logic enabling the server system to identify and respond to new connection requests at a different level of priority than a priority assigned to requests associated with existing connections.

13. A method for allocating received requests in a multiprocessor network server including a network interface, an intelligent switch, a default handler processor, and a set of specialized handler processors, the method comprising the steps of:

receiving, by the network interface, a message packet including a request;

passing at least the request to the intelligent switch;

determining the request is a new request, and in response performing the further steps of:

identifying by the default handler processor, based upon a request type of the new request, a handler processor from the set of specialized handler processors that is capable of executing at least a portion of the new request, and

reassigning by the default handler processor, the new request to the identified handler processor to perform at least a portion of the new request, wherein the intelligent switch creates a request table entry identifying the request and the identified handler processor to which at least of portion of the new request is reassigned; and

executing, by the identified handler processor, at least the portion of the new request, wherein during the executing step the identified handler processor communicates with the network interface via the intelligent switch; thereby bypassing the default handler processor while executing at least the portion of the new request.

14. The method of claim 13 wherein a storage server system is linked to the intelligent switch via a non-blocking switch, and the executing step comprises transferring data from the storage server to the network interface.

15. The method of claim 14 further comprising arranging stored content within the storage server system as a set of version controlled partitions.

16. The method of claim 15 further comprising incorporating a straddle into a partition, thereby facilitating continuous availability of all stored data while a particular partition is relocated within the storage server system.

17. The method of claim 14 further comprising maintaining, by the storage server system, a state of a file in the form of a bitmap entry; and wherein a first bit is associated with a creator of new data in the file and a second bit is associated with a deleter of data stored in the file.

18. The method of claim 14 further comprising receiving, by the intelligent server, a message from the non-blocking switch in the form of ATM cells.

19. The method of claim 14 further comprising providing, within the set of specialized handler processors, a processor facilitating transfer of a file stored on the storage server system.

20. The method of claim 19 wherein the processor facilitating transfer of a file operates in accordance with a TCP named file transfer protocol over an identified connection.

21. The method of claim 14 further comprising providing, within the set of specialized handler processors, a processor including functionality facilitating transforming the data within a file prior to transfer.

22. The method of claim 13 further comprising providing, within the set of specialized handler processors, a processor including computer gateway interface (CGI) functionality.

23. The method of claim 13 further comprising storing data retrieved from a data storage facility in a data retrieval buffer interposed between a data storage facility and the set of specialized handler processors, the data retrieval buffer being independently accessible with respect to a primary RAM utilized by the default handler processor.

24. The method of claim 13 further comprising differentiating a new connection request from a request associated with an existing connection, thereby facilitating assigning a first priority to the request associated with the existing connection and a second priority to the new connection request.